Rev. BASIC

Checkout Launch and Control System (CLCS)

System Test Plan

March 12, 1997

Prepared By:

CLCS System Test Team
CLCS Technical Integration
DE-CLC
Kennedy Space Center, FL 32899

Checkout Launch and Control System (CLCS)

System Test Plan

Prepared By:

____
David Reinhardt, System Test Engineer

____
Timothy McKelvey, System Test Engineer

Approved By:

____
Arnold Postell, Technical Integration Lead

____
Kirk Lougheed, System Engineering and
Technical Integration Lead

____
Retha Hart, CLCS Program Manager

____
Tom Swanson, NASA Safety & Mission Assurance

Table of Contents

1.0 Scope

This document defines the system test plan for use in the development and delivery of the Checkout and Launch Control System (CLCS).

1.1 Identification

This document is the Checkout and Launch Control System (CLCS) System Test Plan Document, Rev. BASIC

1.2 Purpose

The purpose of this document is to define the plan to be used by CLCS Systems Integration and Test engineers to ensure that the CLCS is thoroughly tested at the system level. As the development of the CLCS progresses, this document will be updated as required.

1.3 System Overview

The CLCS is composed of a Real Time Processing System (RTPS) and a support system called the Shuttle Data Center (SDC). The RTPS provides the capability to checkout and control the elements of the current Shuttle Transportation System (STS) and Ground Support Equipment (GSE). It provides support to the Space Shuttle Program into the 21st Century and a basic infrastructure upon which to base future design projects such as the Orbiter Upgrades and X-33/RLV.

The CLCS replaces the current Launch Processing System (LPS) with state-of-the-art Commercial Off the Shelf (COTS) based technology. Wherever possible, COTS software is used instead of developing software where the requirements of the task are satisfied by the COTS software products. Any developed software is written in high level languages which have demonstrated a high degree of portability between platforms. COTS hardware is also utilized as much as possible in the CLCS. This strategy provides a reliable system that is both supportable, with only minimal software and hardware upgrade impacts, and expandable, with a solid base design.

1.4 Document Organization

This document is divided into three sections and four appendices:

Section 1, Scope, discusses the purpose of the CLCS System Integration Operations Test, provides a system overview , and describes software and hardware configurations for the system.

Section 2, Applicable Documents, lists the documents used to create and supporting this document.

Section 3, Test Plan Description, contains a detailed description of the CLCS approach to system level testing

Appendix A, Acronyms and Definitions, contains a listing of acronyms and selected word definitions (for words which may have multiple interpretations)

2.0 Applicable Documentation

The following documents of the revision shown form a part of this document to the extent specified.

2.1 Parent Documents

The documents in this paragraph establish the criteria and technical basis for the existence of this document. The parent documents are:

- Program Management Plan
- System Engineering Management Plan (SEMP)

2.2 Applicable Documents

Applicable documents are those documents which form a part of this document. These documents, at the revisions listed below, carry the same weight as if they were stated within the body of this document.

- Verification Plan
- CM Plan
- Integration Management Plan

2.3 Reference Documents

Reference documents are those documents which, though not a part of this document, serve to clarify the intent and contents of this document.

- System Level Specification
- Concept of Operation
- System Design Specification

3.0 Test Plan Description

3.1 CLCS Test Philosophy

3.1.1 CLCS Test Approach Overview

This section provides an overview of all testing in the development of the CLCS. While this test plan covers only system testing (referred to as System Integration/Operations Tests or SIOT), it is important to understand the SIOT in the context of the overall development test flow.

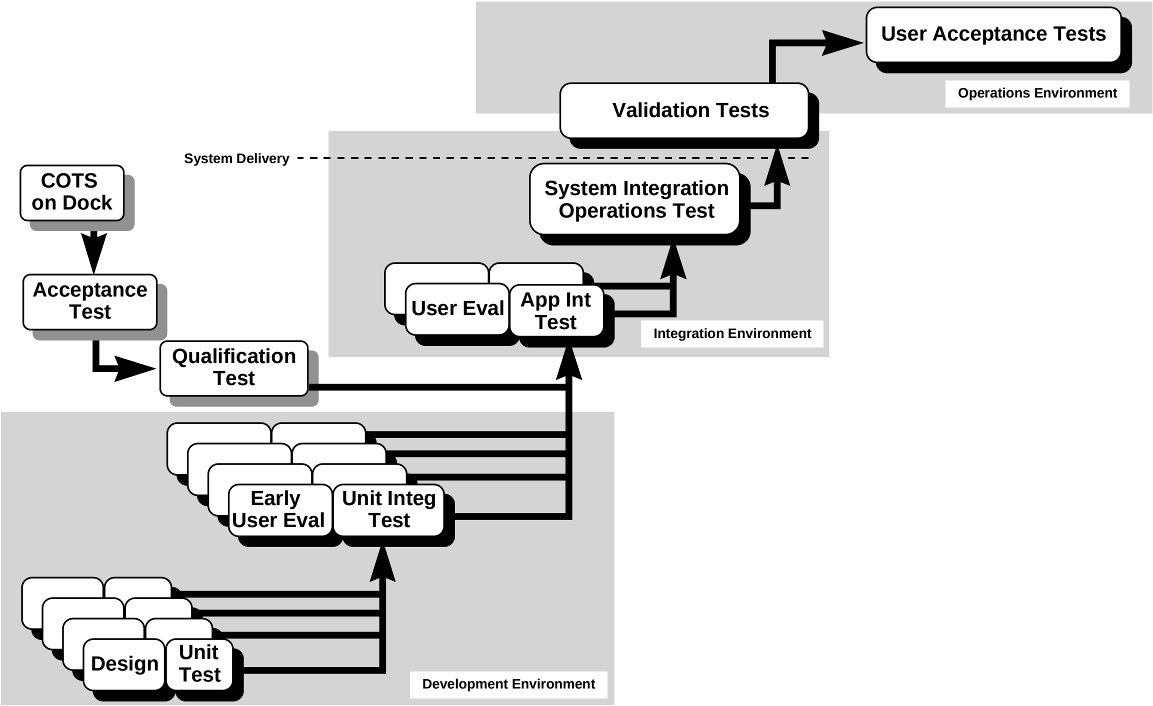


Figure 1. CLCS Development Test Approach (One Delivery Cycle Shown)

CLCS development testing starts at the lowest level (Unit Tests and Acceptance Tests) and goes through System Integration and Operation Testing. Validation and User Acceptance Testing are performed outside of the development organization and are key parts of the system certification process. Basic guidelines for all levels of development testing include the following:

- Each test is developed and reviewed in relation to previous testing
  - Unit Integration Tests (UIT's), Application Integration Tests (AIT's), and SIOT's do not simply repeat procedures performed in previous tests. Emphasis is placed on what needs to be tested in the context of each level of test

- Test procedures or procedure segments from previous tests (at the same level) are modified as necessary and reused where possible, e.g., tests from a given SIOT may be used in a subsequent SIOT.

- Early user evaluation will be an integral part of the test development process
  - Feed back from user experience with early versions of software will allow for the development of more focused tests at the UIT level and above
  - Areas of concern to the user community will be emphasized
  - Early evaluation will facilitate coordination of the content of different levels of testing

- End-to-end testing process will be consistent across all software types and development organizations:
  - Re-engineered "GOAL type" applications
  - System/Platform Services
  - application services
  - NASA
  - USA
  - LMSMSS

- Test procedures will be carefully reviewed to ensure that people, hardware and software are not jeopardized during testing.

- There will be fully documented and approved test procedures for each level of testing, test documents will be controlled under the CLCS project CM process.

3.1.2 CLCS Test Definitions

3.1.2.1 Unit Test

Unit Tests (UT's) are performed by the software developer to verify basic functionality of a demonstrable segment of a CSCI; for example, a single program that can be meaningfully executed in a "stand alone" mode. Unit tests require a minimal number of stubbed interfaces and are tested against Functional Requirements . This is the first level of testing for original software, and may be omitted for more mature software (e.g., reuse code, incremental update to code that had previously been Unit Tested). The level of formality (amount and type of documentation) is minimal. The tests are signed off by the development lead. QA will review the test procedures and may or may not witness/signoff the tests. This will be determined jointly by the development lead and QA on a case by case basis. Number of tests and test cases for UT's are finalized at Design Panel 3. Unit tests are performed in the Development Environment.

3.1.2.2 Unit Integration Test

Unit Integration Tests (UIT's) are performed by the software developer to verify basic functionality and successful integration of a set of programs (CSC) with its new and/or modified units. The UIT's are tested against Functional Requirements and can be the first level of testing for reuse software. UIT's are witnessed and signed-off by development lead and QA. UIT test plan data are finalized at Design Panel 3. UIT's are performed in the Development Environment.

3.1.2.3 Acceptance Test

Acceptance Tests (AT's) are performed by the procuring organization to verify that COTS hardware and/or software meet the requirements of the procurement specification. This is the first level of testing for COTS hardware and software, and is roughly equivalent to a Unit Integration Test.

3.1.2.4 Application Integration Test

Application Integration Test's (AIT's) are performed by the software developer and/or thread lead to verify basic functionality and successful integration of a functionally complete set of programs (thread or CSCI's, e.g., GLS, LOX, System Services) with its new and/or modified units in an operational environment. AIT's are the final tests against Functional Requirements and are the final level of testing prior to system tests (SIOT). The tests are reviewed by the development lead, thread lead, QA and System Integration & Test, and are witnessed and signed-off by development/thread lead and QA. The number of tests and test cases for AIT's are finalized at Design Panel 2. AIT's are performed in the Integrated Development Environment (IDE), which is configured the same as the Operational environment and is used as a dedicated system test and integration facility.

3.1.2.5 Qualification Test

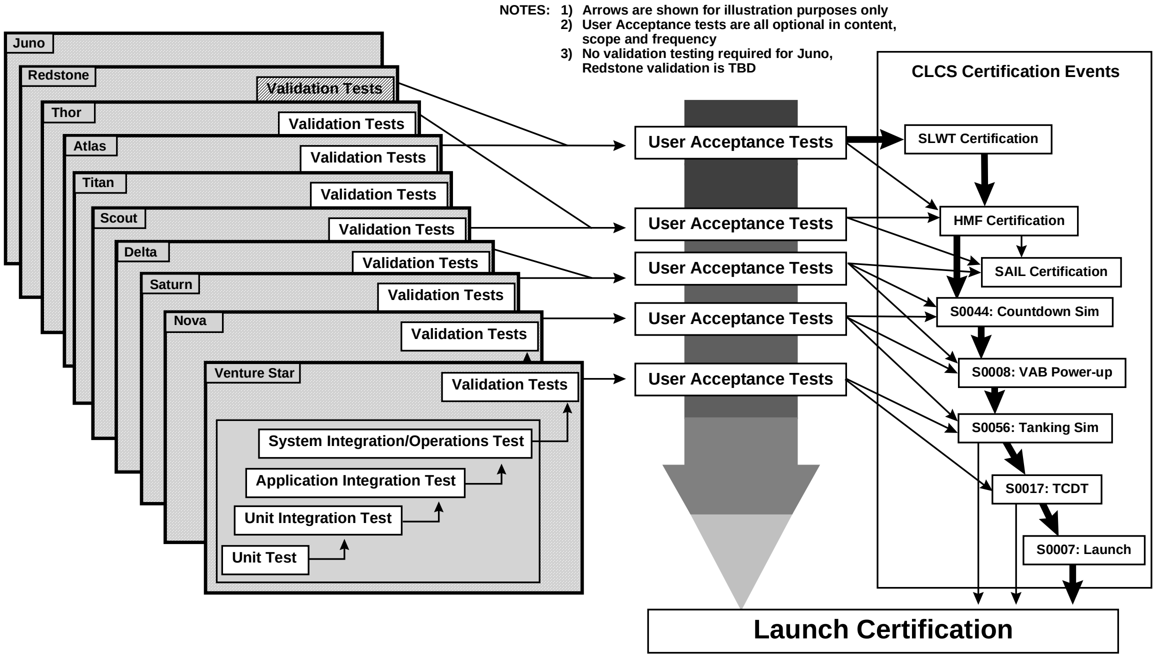
Qualification Tests (QT's) are performed by the procuring organization to verify that COTS hardware and/or software meet the CLCS platform requirements. This is the final level of standalone testing for COTS hardware and software, and is roughly equivalent to an Application Integration Test.

3.1.2.6 System Integration and Operations Test

The System Integration/Operations Tests (SIOT's), are performed by the System Test team (which is a subset of the System Integration and Test group) to verify successful integration of a system delivery and to demonstrate that the system is "operable" in all required modes (development, maintenance, system operations, test/launch support, etc.). The SIOT's validate system level requirements and external interfaces emphasizing system functionality, performance and system wide data flow. They typically will include regression testing for previously delivered capabilities. The SIOT's are tested against System Level Specification Requirements and are reviewed by QA, System Integration & Test, development/thread leads and witnessed and signed-off by System Integration & Test and QA. The number of tests and test cases for SIOT's are finalized after the last Design Panel 3 for a given delivery and are based on threads, operational scenarios and operations constraints. This is the final level of testing prior to putting the system in an operations environment for user certification/validation.

3.1.3 CLCS System Validation and Certification Process

The CLCS validation and certification process is employed by the end user community to verify that the development organization has delivered a functionally complete and correct system which can be used as safely and reliably as possible to support hazardous operations. This process is under the control of the end user community and is not the responsibility of the System Integration and Test team. It is presented here for information purposes only. The definitive source for the validation and certification process is the *<TBD Document>*



NOTES: 1) Arrows are shown for illustration purposes only
2) User Acceptance tests are all optional in content, scope and frequency
3) No validation testing required for Juno, Redstone validation is TBD

Figure 2. Test and Certification Philosophy

3.1.3.1 Validation Tests

Validation tests are performed by an independent group at KSC and are intended to verify that the product delivered to the operations organization from the development organization meets the users requirements. Validation tests are executed for each delivered system They occur in the Integrated Development Environment or the Operations environment as appropriate (determined by the testing organization). Validation tests are supported by development/thread leads and Systems Integration & Test.

3.1.3.1 User Acceptance Tests

User Acceptance Tests (UAT's) verify that the system is processing data correctly (data and displays are valid) and that the system can be certified for use in approved types of operations (e.g., SLWT, HMF, Launch). The scope of the acceptance tests is defined by user community on a per test basis. User acceptance tests are executed in the operations environment as required by users (not necessarily at each delivery), usually in preparation for operational certification. User acceptance tests are supported by development/thread leads and Systems Integration & Test. This is the final level of test prior to full operations support.

3.2 CLCS Testing Procedures and Ground Rules

3.2.1 Test Flow

System tests for each CLCS delivery will, where practicable, conform to the following schedule template:

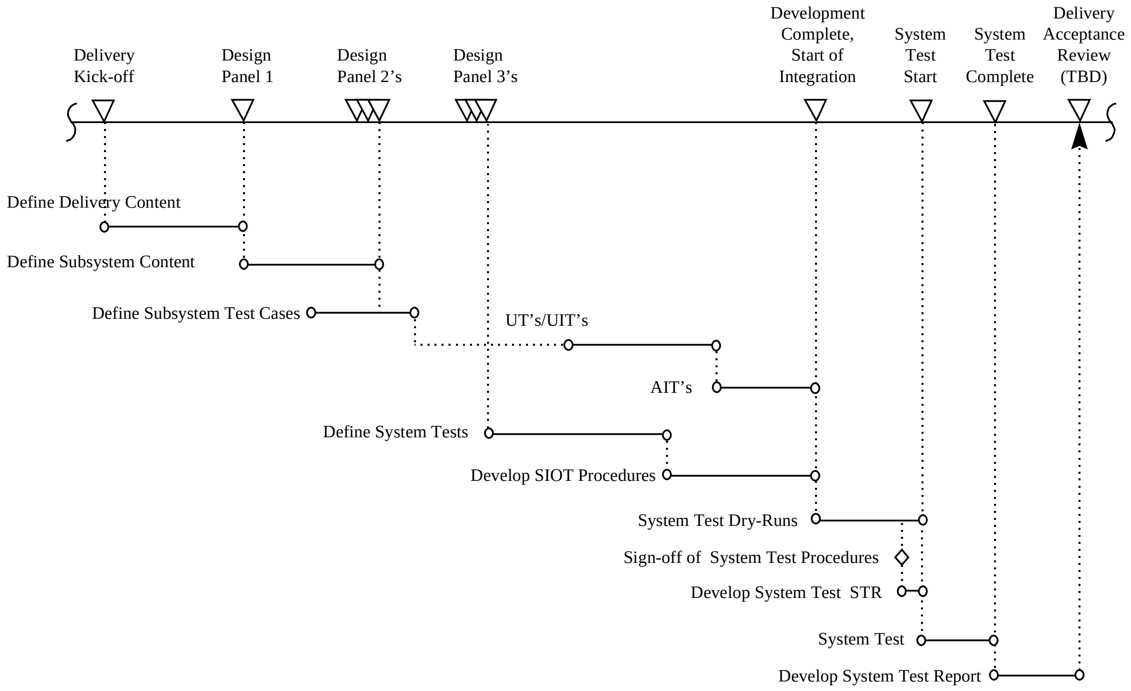


Figure 3. Test Development Flow

Development of SIOT procedures begins after the last Design Panel 3 for a given delivery, though the System test team will participate in the prior design reviews. At this time, the test cases will be defined and then approved by the Technical Integration Lead with concurrence of the System Engineering & Integration leads and the User Liaison. Procedure development will be done incrementally starting with test case development and finishing with the dry-runs prior to final approval of the procedures and execution of the test. Subsystem level testing will be taken into account, as will inputs from early user evaluations of the delivery.

3.2.1 System Test Reviews and Documents

3.2.1.1 Start of Test Review

The Start of Test Review (STR) is presented, if deemed necessary by the Test Conductor, prior to the start of a system test. The purpose of the presentation is to describe the state that the system is in and whether or not it is ready for testing.

The STR is chaired by the NASA CLCS Technical Integration lead and is presented by the CLCS System Test team. The goal of the review is to obtain concurrence from the NASA CLCS Technical Integration lead and from QA that the system is ready for system level testing. This concurrence provides authority to commence testing. The STR presentation includes the following information:

- Delivery content summary
- Target system configuration for the delivery
- System configuration to be used during the test
- Summary of previous testing (subsystem and system level tests)
- Summary of dry run testing
- Known problems
- Ground rules for test conduct
  - Roles and responsibilities
  - Handling test exceptions, including emergency procedures as necessary
  - Problem reporting procedures
- Test schedule and locations

3.2.1.2 SIOT Test Procedures

The SIOT Test Procedures Documents will define:

- Test scope and intent
- Test cases and test dependencies
- Tracing of system level requirements to test cases
- Specific test procedures and sign off

3.2.1.3 Daily Test Progress Review

If necessary, in the opinion of the Test Conductor or the QA witness, a daily Test Progress Review (TPR) presentation will be provided by the Test Conductor at the beginning of each day of testing. This review will cover test progress to date, summary of problems found during the previous day(s) of testing and tests planned for that day.

3.2.1.4 Test Results Review

At the completion of system testing, a Test Results Review (TRR) will be held (at the option of the Test Conductor and/or the QA witness). This review will summarize the testing completed, problems found during testing and any follow up actions required to declare the successful completion of the test.

3.2.1.5 Test Report

The Test Report (TR) will be released after the completion of all system testing and will be included in the delivery package for the given system delivery. This report will include a summary of the test procedures, the test results, problems found and a copy of the “as run” test procedures, including any additions or modifications to the procedures generated during the test.

This report is the System Test organization’s input to the Delivery Acceptance Review (DAR). The DAR is a meeting, chaired by the CLCS Delivery Manager, at which the delivery package is reviewed and the Delivery is declared complete. The delivery package includes such things as magnetic tape copies of software to be delivered, all development documentation (users guides, requirements, design documents), all test documentation (subsystem and system level tests) and known problems.

3.2.2 Roles and Responsibilities

The following matrix shows the roles and responsibilities for each organization involved in the CLCS development. Following that are the detailed descriptions of those responsibilities.

Organization	SIOT Test Plan	SIOT Test Procs.	SIOT Dry Run	STR	TPR	Test Results Review	Test Report
CLCS Developers	R	R	S				
CLCS System Integration	R	R	S,E	W		W	R
CLCS System Test	G	G,E	E	G,E	G,E	G,E	G
CLCS System Admin		R,S	S	S	S	S	
NASA QA	R,A	R,W,A		R,A	R,A	R,A	R,A
USA QA	R	R					R
LMSMSS QA	R	R					R
End User Community	R	R,S					
Operations Support		S	S	S	S	S	

W=witness, E=execute, R=review, A=approve, S=support, G=generate

3.2.2.1 CLCS Development Organization Responsibilities

CLCS Developers - Review and support development of system test procedures. This includes the development of test data, scripts and procedures, as well as providing documentation of previous subsystem level testing and which functional requirements have been met. Support dry runs and support test execution as necessary.

CLCS Systems Integration - Review and support development of system test procedures, ensure that any required pre-test system builds have been completed, support dry runs and support test execution as necessary (e.g., problem isolation during system test execution).

CLCS System Test - Generate all system test documents, execute test and perform all reviews. Act as Test Conductor. The Test Conductor is in charge of the execution of a SIOT. The test conductor is responsible for ensuring that the system is in the proper configuration for the test, that all required approvals have been obtained and that the test support personnel and tools are in place. The Test Conductor is the final authority as to the conduct of the test, including changes to the test procedures, documenting problems found and changes in the system configuration during the test.

CLCS System Administration - Certain test procedures will require that the tester have the ability and permission to use administrative tools and procedures ("root" access). The system administrators will support the development and execution of these tests.

3.2.2.2 Quality Organization Responsibilities

NASA Quality - NASA Quality representatives will approve test plans and procedures, witness and approve tests, approve any changes to test procedures and configurations and have the authority to stop testing if significant problems are found.

USA Quality - Support NASA Quality as required.

LMSMSS Quality - Support NASA Quality as required.

3.2.2.3 End User Community Responsibilities

The end user community is expected to review test cases and support their development. They also may be requested to support test procedure development to ensure conformance with operational concepts, scenarios and plans.

3.2.2.4 Other Support Personnel Responsibilities

Operations Support - Certain test procedures will require the use of operational facilities and capabilities such as hardware configuration/reconfiguration and accessing live or simulated data. The operations support organization will support the development and execution of these test procedures as necessary to support a given test configuration.

3.2.3 Test Exception Procedures

In the event of test exceptions (i.e., test results that do not match the expected results), the Test Conductor has the responsibility to see that the appropriate problem reports are generated, properly classified regarding impact on both the delivery and on the continuation of the test, and properly classified regarding urgency. The Test Conductor will, if warranted in his/her opinion, suspend testing and schedule the resumption of the test as soon as possible. Otherwise, the PR will be generated, classified and submitted, testing will resume and the PR will be reported as required in the TPR, TRR, TR and the CLCS QE Plan.

3.2.4 Test Change Procedures

The Test Conductor has the sole authority to add to, delete from or red-line the test procedures during the execution of the test. All changes must be approved by the QA witness. If, in the opinion of the Test Conductor or the QA representative, the changes are excessive or have the potential to endanger personnel, hardware or software, a problem report can be written against the test procedures. In this case, the test would proceed as written as much as practicable and the updated test procedures would be executed in a separate test which in turn would close the PR.

3.2.5 Problem Report Definitions

Informal problem reports generated during development and still open at the time of the system test will be converted to the formal problem reporting system used for CLCS at the time of system delivery. The following definitions will be used for problem reports found prior to and during system testing. Note that these may not be the same definitions as those used in an operational environment:

- **Critical:** A critical problem is one in which a user (or the system on behalf of a user) is unable to perform a required task, and no viable workaround exists. Problems involving safety are considered to be critical.

- **Major:** A major problem is one in which a user is unable to perform a required task, but a workaround is available, even though the workaround may impact user functions/operations.

- **Minor:** A minor problem is one with minimal impact to the users, a workaround is available and does not significantly affect the ability of the users to perform their function.

The above PR classification refers only to functionality. Urgency in obtaining a correction to the problem is not considered in this classification.

Appendix A - Acronyms and Definitions

AIT	Application Integration Test
AT	Acceptance Test - Test to accept hardware and software from a vendor
CLCS	Checkout and Launch Control System
Certification	Final approval to use a system for a specified set of operations (e.g., hazardous operations in the HMF, launch operations, etc.)
CM	Configuration Management
COTS	Commercial Off The Shelf
DAR	Delivery Acceptance Review
EDL	Engineering Development Laboratory
GSE	Ground Support Equipment
HCI	Human Computer Interface
HMF	Hypergol Maintenance Facility
HW	Hardware
IDE	Integrated Development Environment
I/F	Interface
KSC	Kennedy Space Center
LAN	Local Area Network
LCC	Launch Control Complex
LMSMSS	Lockheed Martin Space Missions Systems and Services
LPS	Launch Processing System
NASA	National Aeronautics and Space Administration
OS	Operating System
PR	Problem Report
QA	Quality Assurance
QE	Quality Engineering
QT	Qualification Test
RLV	Reusable Launch Vehicle
RTPS	Real Time Processing System

RVM	Requirements Verification Matrix
SDC	Shuttle Data Center
SDE	Satellite Development Environment
SEMP	System Engineering Management Plan
SIOT	System Integration/Operations Test
SLWT	Super Light Weight Tank
STR	Start of Test Review
STS	Shuttle Transportation System
SW	Software
TC	Test Conductor
TPR	Test Progress Review
TR	Test Report
TRR	Test Results Review
UAT	User Acceptance Test - Test performed by user community post delivery as part of the system certification process
UIT	Unit Integration Test
USA	United Space Alliance
UT	Unit Test
Validation	Testing performed by organization(s) outside of the developing organization to ensure that the delivered system processes data correctly and conforms to the operations concepts